

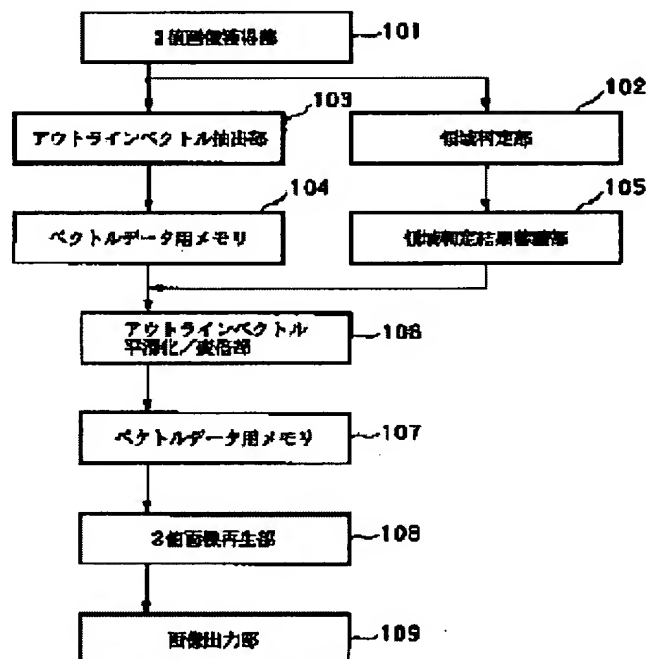
# PICTURE PROCESSING METHOD AND PICTURE PROCESSOR

**Patent number:** JP8263650  
**Publication date:** 1996-10-11  
**Inventor:** YOSHITANI AKIHIRO; SHIGEE NOBUYUKI  
**Applicant:** CANON KK  
**Classification:**  
**- International:** G06T3/40; G06T5/20; G09G5/36; H04N1/393;  
H04N1/40; G06T3/40; G06T5/20; G09G5/36;  
H04N1/393; H04N1/40; (IPC1-7): G06T5/20; G06T3/40;  
G09G5/36; H04N1/393; H04N1/40  
**- european:**  
**Application number:** JP19950065986 19950324  
**Priority number(s):** JP19950065986 19950324

Report a data error here

## Abstract of JP8263650

**PURPOSE:** To obtain a clear variable power picture without the deterioration of picture quality by setting only a starting point considered to be a non-salient point to be the object of a smoothing processing.  
**CONSTITUTION:** Outline vector data and a corresponding area judgement result are respectively read from a memory for vector data 104 and an area judgement result accumulation part 105. The number of picture elements judged to be a pseudo half-tone area in an area near the starting point of the outline vector is counted, and the number of pictures is stored in a variable C. When the number of picture elements, which is stored in the variable C, is less than a prescribed number (k), the starting point of the outline vector is registered as the non-salient point. When the number of the picture elements, which is stored in the variable C, is more than the prescribed number (k), on the other hand, the starting point of the vector is registered as a salient point. Thus, whether a processing on the whole outline vectors stored in the memory for outline vector data 104 is terminated or not is judged. When the processing on the whole outline vectors is terminated, the smoothing processing is executed.



Data supplied from the esp@cenet database - Worldwide

3/3

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-263650

(43) 公開日 平成8年(1996)10月11日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 T 5/20			G 0 6 F 15/68	4 1 0
			G 0 9 G 5/36	5 2 0 P
G 0 9 G 5/36	5 2 0	9377-5H	H 0 4 N 1/393	
H 0 4 N 1/393			G 0 6 F 15/66	3 5 5 L
			H 0 4 N 1/40	Z
			審査請求 未請求 請求項の数8 O L (全 8 頁)	

(21) 出願番号 特願平7-65986

(22) 出願日 平成7年(1995)3月24日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 吉谷 明洋

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72) 発明者 重枝 伸之

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

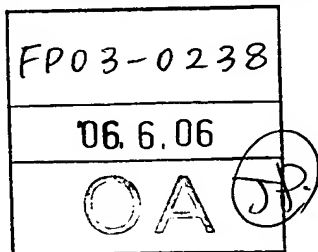
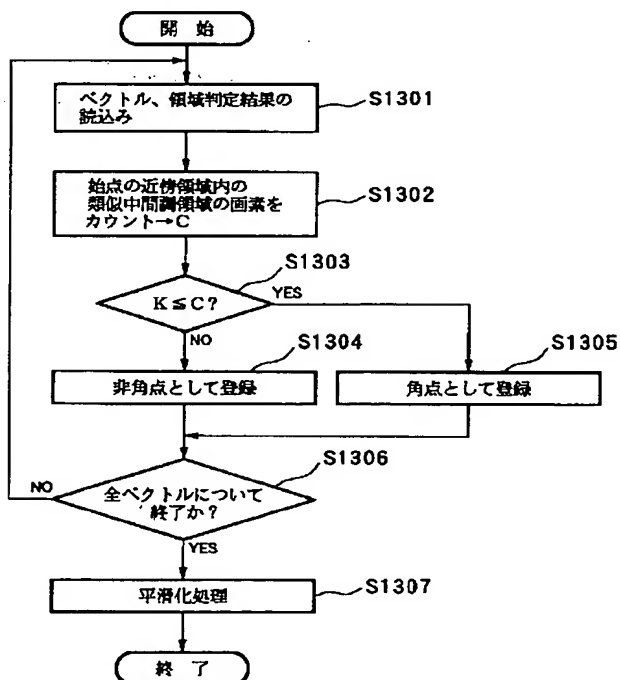
(74) 代理人 弁理士 大塚 康徳 (外1名)

(54) 【発明の名称】 画像処理方法及び画像処理装置

(57) 【要約】

【目的】疑似中間調処理を施された画像の輪郭の平滑化処理における画質劣化を防止する。

【構成】ステップS1301において、ベクトルデータ及びそれに対応する領域判定結果を夫々読む。ステップS1302において、読出されたベクトルの始点の近傍領域内の疑似中間調領域と判定された画素数をカウントし、その画素数を変数Cに格納する。ステップS1303において、変数Cに格納された画素数と所定数kとを比較し、変数Cの値が所定数k未満である場合は、ステップS1304に進み、ベクトルの始点を非角点として登録する。一方、変数Cの値が所定数k以上である場合は、ステップS1305に進み、ベクトルの始点を角点として登録する。以下、ステップS1306において、全ベクトルについて処理が終了したと判定したら、ステップS1307に進み、平滑化処理を実行する。



## 【特許請求の範囲】

【請求項1】 画像の輪郭ベクトルを抽出する抽出工程と、  
前記画像の疑似中間調領域を識別する識別工程と、  
前記抽出工程において抽出された前記輪郭ベクトルの始点が前記疑似中間調領域内に位置するか否かを判定する判定工程と、  
前記判定工程による判定結果が、前記疑似中間調領域内に前記始点が位置しない旨を示すときに、前記輪郭ベクトルを平滑化する平滑化工程と、  
を具備することを特徴とする画像処理方法。

【請求項2】 前記識別工程は前記始点の周辺領域における前記疑似中間調領域を識別することを特徴とする請求項1記載の画像処理方法。

【請求項3】 前記判定工程は前記始点の近傍領域内における前記疑似中間調領域を構成する画素数が所定数に満たない場合に、前記始点が前記疑似中間調領域内に位置しない旨を判定する工程であることを特徴とする請求項1または請求項2記載の画像処理方法。

【請求項4】 前記平滑化工程において加工された前記画像を変倍して変倍画像を生成する生成工程を更に具備することを特徴とする請求項1乃至請求項3のいずれか1項に記載の画像処理方法。

【請求項5】 画像の輪郭ベクトルを抽出する抽出手段と、  
前記画像の疑似中間調領域を識別する識別手段と、  
前記抽出手段によって抽出された前記輪郭ベクトルの始点が前記疑似中間調領域内に位置するか否かを判定する判定手段と、  
前記判定手段による判定結果が、前記疑似中間調領域内に前記始点が位置しない旨を示すときに、前記輪郭ベクトルを平滑化する平滑化手段と、  
を具備することを特徴とする画像処理装置。

【請求項6】 前記識別手段は前記始点の周辺領域における前記疑似中間調領域を識別することを特徴とする請求項5記載の画像処理装置。

【請求項7】 前記判定手段は前記始点の近傍領域内における前記疑似中間調領域を構成する画素数が所定数に満たない場合に、前記始点が前記疑似中間調領域内に位置しない旨を判定する手段であることを特徴とする請求項5または請求項6記載の画像処理装置。

【請求項8】 前記平滑化手段によって加工された前記画像を変倍して変倍画像を生成する生成手段を更に具備することを特徴とする請求項5乃至請求項7のいずれか1項に記載の画像処理装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、画像処理方法及び画像処理装置に係り、特に画像を平滑化するための画像処理方法及び画像処理装置に関する。

## 【0002】

【従来の技術】 2値画像からアウトライン・ベクトルを抽出し、そのアウトライン・ベクトルを用いて画像の変倍・平滑化を行うことにより美しい変倍画像を得る手法が提案されている。この手法は、アウトライン・ベクトル抽出、ベクトル平滑化／変倍、2値画像再生の3段階の処理から構成される。以下、夫々の処理について図を参照しながら説明する。

【0003】 先ず、アウトライン・ベクトル抽出処理について図2を参照しながら説明する。図2の(a)はラスタ形式で入力された入力画像である。この入力画像を3×3のウィンドウで走査し、そのウィンドウの中心である注目画素とその周りの画素の状態を調べ、相隣り合う2つの画素の画素値が異なる場合には、図2の(b)のように、その2つの画素間に素輪郭ベクトルを検出し、検出した素輪郭ベクトルを、図2の(c)のように黒画素を囲むようなループの形に整列して、アウトラインベクトルとする。ベクトルの方向は、ベクトルの始点から終点に向かって右側に黒画素が存在するように定義する。

【0004】 次に、ベクトル平滑化／変倍処理について説明する。このベクトル平滑化／変倍処理は第1平滑化／変倍処理、及び第2平滑化処理の2段階の処理に分けられる。

【0005】 先ず、第1平滑化／変倍処理について説明する。この処理は、先ず、前述のアウトライン・ベクトル抽出処理で抽出され、ループ単位に整列されたベクトル群に対して、あるベクトルを注目ベクトルとして、その前後数本のベクトルの向き、長さの組合せにより場合分けを行う。次に、その場合分けの結果により、各ベクトルの始点（以下、輪郭点とも表現する）が原画像（入力されるラスタ画像データをスキャナでスキャンする前の画像）において「角」であったかどうかを判断し、「角」であると判断されればその輪郭点を「角点」とし、その情報をその輪郭点に対するフラグとして別に記憶する。一方、「角」でなければ輪郭線が滑らかになるように輪郭点を再設定することにより平滑化を行う。ここで、「角」とは、例えば多角形の原画像の頂点等のように平滑化処理の対象とならない部分、或いは平滑化処理の対象とすべきでない部分をいい、また、「角点」とは、「角」を示す点をいう。

【0006】 図3乃至図5は平滑化処理の対象となるラスタ画像データの例を示す図である。ラスタ画像データが図3の(a)のように短いベクトルが階段状に並んでいる場合は、このベクトルの部分が原画像において滑らかな斜めの線の一部であると解釈して、階段が滑らかになるように図3の(b)のように輪郭点座標を再設定する。

【0007】 また、ラスタ画像データが図4の(a)のように長いベクトルが2本隣接している場合は、その2

本のベクトルの縫ぎ目の点（図4の（b）のP401）を角点として記憶し、座標値の再計算は行わず、そのまま輪郭点とする。

【0008】また、ラスタ画像データが図5の（a）のような1ドットの大きさのノッチ（出張りまたはへこみ）があるような場合は、これを除去して図5の（b）のように処理する。即ち、ノッチの部分の輪郭点P501乃至P504はすべて削除される。

【0009】以上の処理によって得られた再設定後の各輪郭点の座標値に、縦方向、横方向の所望の倍率を乗ずることにより、変倍処理を行う。

【0010】次に、第2平滑化処理について説明する。第2平滑化処理は、先の第1平滑化処理によって得られた各輪郭点のデータを入力データとし、第1平滑化処理において角点であると判断されなかった点（非角点）について、その周辺のベクトル群がなお一層滑らかになるように、再度座標値の計算を行う。具体的には、その座標値と、左右両隣の輪郭点の座標値との加重平均を計算し、その値をその点の新しい座標値とする。

【0011】次に2値画像再生処理について説明する。この2値画像再生処理は、先に述べたベクトル平滑化／変倍処理によって得られたベクトルデータを2値画像として描画し、ループ内を黒画素で埋める処理を行うことにより、2値画像を再生する処理である。

【0012】以上のような処理を行うことにより、美しい変倍画像を得ることができる。例えば、図6のような入力画像データが与えられた場合には、まずアウトラインベクトル抽出処理によって、図7のようにベクトル（輪郭点P601乃至P608）が抽出される。次に、第1平滑化処理によって、図8のように輪郭点が再設定されて角点、非角点の区別がつけられた後に変倍処理が行われる。尚、図8において、P611、P612、P617、P618、P619は角点、P613乃至P616は非角点を示している。更に、第二平滑化処理で非角点に対し隣の点の座標値との加重平均をとることで図9のように輪郭線が一層滑らかになるようにする。尚、図9において、P611乃至P619は、第2平滑化処理前、P611、P612、P621、P622、P623、P624、P617、P618、P619は第2平滑化処理後の輪郭点である。図10は第2平滑化処理後に2値画像再生処理を行った結果、図11は上記一連の処理を施さない場合の結果である。図10及び図11の比較から明らかなように、アウトラインベクトル抽出処理、平滑化処理等を施すことにより、いわゆるジャギーの目立たない変倍画像を得ることができる。

【0013】

【発明が解決しようとする課題】しかしながら、上記従来例においては、アウトライン変倍処理を一般の2値画像に適用する場合、例えば入力画像が疑似中間調処理を施された画像である場合等に、ノッチ除去などの平滑化

処理において入力画像の局所的な濃度を保存しないために、変倍後の画像の画質が劣化するという問題点を有している。

【0014】本発明は、上記問題点に鑑みなされたものであり、疑似中間調処理を施された画像の輪郭の平滑化処理における画質劣化を防止することを目的とする。

【0015】

【課題を解決するための手段】及び

【作用】上記問題点を解決するため、本発明に係る画像処理方法は、画像の輪郭ベクトルを抽出する抽出工程と、前記画像の疑似中間調領域を識別する識別工程と、前記抽出工程において抽出された前記輪郭ベクトルの始点が前記疑似中間調領域内に位置するか否かを判定する判定工程と、前記判定工程による判定結果が、前記疑似中間調領域内に前記始点が位置しない旨を示すときに、前記輪郭ベクトルを平滑化する平滑化工程とを具備することを特徴とし、疑似中間調処理を施された画像の輪郭の平滑化処理における画質劣化が防止される。

【0016】また、本発明に係る画像処理装置は、画像の輪郭ベクトルを抽出する抽出手段と、前記画像の疑似中間調領域を識別する識別手段と、前記抽出手段によって抽出された前記輪郭ベクトルの始点が前記疑似中間調領域内に位置するか否かを判定する判定手段と、前記判定手段による判定結果が、前記疑似中間調領域内に前記始点が位置しない旨を示すときに、前記輪郭ベクトルを平滑化する平滑化手段とを具備することを特徴とし、疑似中間調処理を施された画像の輪郭の平滑化処理における画質劣化が防止される。

【0017】

【実施例】

【第1の実施例】以下、本発明の好適な第1の実施例を説明する。図1は、本実施例に係る画像処理装置の構成を示すブロック図である。図1において、2値画像獲得部101は、処理対象となる2値画像を獲得して、ラスタ走査形式の2値画像データをアウトライン抽出部103及び領域判定部102に対して出力する。2値画像獲得部101は、例えば、ファクシミリ装置における電話回線から画像を取り込む入力インタフェース等と同様の機能であれば良い。

【0018】領域判定部102は、入力画像に対して公知の方法を用いて、画像中の黒画素が文字画像領域にあるか疑似中間調画像領域にあるかを判断し、その結果を画素単位で領域判定結果蓄積部105に対して出力する。

【0019】領域判定結果蓄積部105は、領域判定部102から供給される各画素についての領域判定結果を蓄積するための入力画像の一画面分に相当する容量を有するメモリである。

【0020】アウトライン・ベクトル抽出部103は、入力画像の輪郭線をベクトルとして表したアウトライン

・ベクトルを上述の方法で抽出し、そのアウトライン・ベクトルをベクトルデータ用メモリ104に出力する。

【0021】ベクトルデータ用メモリ104は、アウトライン・ベクトル抽出部103において抽出されたアウトライン・ベクトルデータを一時的に保存するFIFO形式のメモリである。

【0022】アウトライン・ベクトル平滑化／変倍部106は、ベクトルデータ用メモリ104からアウトライン・ベクトルデータを受け取り、領域判定結果蓄積部に蓄積されている領域判定結果を逐次参照しながら、後述のアルゴリズムに従ってアウトライン・ベクトルの平滑化・変倍処理を行い、その結果をベクトルデータ用メモリ107に出力する。

【0023】ベクトルデータ用メモリ107は、アウトライン・ベクトル平滑化／変倍部106から出力された、平滑化／変倍処理されたアウトライン・ベクトルデータを一時的に保存するFIFO形式のメモリである。

【0024】2値画像再生部108は、ベクトルデータ用メモリ107から平滑化／変倍処理されたアウトライン・ベクトルデータを読み込み、2値画像をストライプ単位で再生し、処理が終了したら、この再生画像データを画像出力部109へ供給し、画像を出力する。

【0025】次に、アウトライン・ベクトル平滑化／変倍部106で行われるアウトライン・ベクトル平滑化／変倍処理のアルゴリズムについて説明する。このアウトライン・ベクトル平滑化／変倍処理は、第1平滑化処理、第2平滑化処理、変倍処理の3つの処理からなり、第2平滑化処理及び変倍処理については、前述と同様の処理を施す。

【0026】図12は処理対象となる入力画像である。図12において、1200はアウトライン・ベクトル抽出部103によって抽出されたアウトライン・ベクトルの1つである注目ベクトル、1201は注目ベクトルの始点（輪郭点）、1210は、始点1201の近傍領域として定義される領域であって、始点1201を中心とした $N \times N$ の画素から構成される領域である。尚、図12においては、 $N$ の値は4であり、始点1201を中心とした $4 \times 4$ の画素の領域を、始点1201の近傍領域として定義している。また、1220は領域判定部102において黒画素であると判定された画素、1230は領域判定部102において疑似中間調領域であると判定された黒画素である。尚、これらのアウトライン・ベクトル及び領域判定結果は、前述のように、ベクトル用メモリ104及び領域判定結果蓄積部105から得られる。

【0027】以下、これらの情報を元にして実行される第1平滑化処理を図13を参照しながら説明する。先ず、ステップS1301において、ベクトルデータ用メモリ104及び領域判定結果蓄積部105からアウトライン・ベクトルデータ及びそれに対応する領域判定結果

を夫々読込む。

【0028】ステップS1302において、読出されたアウトライン・ベクトルの始点の近傍領域内の疑似中間調領域と判定された画素数をカウントし、その画素数を変数Cに格納する。

【0029】次に、ステップS1303において、変数Cに格納された画素数と所定数 $k$ とを比較する。その結果、変数Cに格納された画素数が所定数 $k$ 未満である場合は、ステップ1304に進み、アウトライン・ベクトルの始点を非角点として登録する。一方、変数Cに格納された画素数が所定数 $k$ 以上である場合は、ステップS1305に進み、ベクトルの始点を角点として登録する。

【0030】ステップS1306において、アウトライン・ベクトルデータ用メモリ104に格納された全アウトライン・ベクトルについて上記処理が終了したか否かを判定する。その結果、全アウトライン・ベクトルについて処理が終了した場合には、ステップS1307に進み、平滑化処理を実行する。一方、全アウトライン・ベクトルについて処理が終了していない場合には、ステップS1301に戻り、残りのアウトライン・ベクトルに関して再度同様の処理を繰り返す。

【0031】尚、ステップS1307における平滑化処理は、従来例における第1平滑化処理と同様の処理である。即ち、ラスタ画像データにおいて図3の(a)のように短いベクトルが階段状に並んでいる場合は、このベクトルの部分が原画像において滑らかな斜めの線の一部であると解釈して、階段が滑らかになるように図3の(b)のように輪郭点座標を再設定する。

【0032】また、ラスタ画像データが図4の(a)のように長いベクトルが2本隣接している場合は、その2本のベクトルの継ぎ目の点（図4の(b)のP401）を角点として記憶し、座標値の再計算は行わず、そのまま輪郭点とする。

【0033】また、ラスタ画像データが図5の(a)のような1ドットの大きさのノッチ（出張りまたはへこみ）があるような場合は、これを除去して図5の(b)のように処理する。即ち、ノッチの部分の輪郭点P501乃至P504はすべて削除される。

【0034】このように、本実施例においては、ステップS1303乃至S1305において、疑似中間調領域であるか否かによって、各輪郭点について角点と非角点に振り分けを行い、その結果、非角点とみなされたアウトライン・ベクトルの始点を、ステップS1307において、図形的な観点から角点と非角点に更に振り分けを行った後に、変倍処理を実行する。

【0035】以上のように、疑似中間調領域であると判断されたアウトライン・ベクトルの始点、即ち、角点とみなされた始点については、平滑化処理の対象とせず、非角点とみなされた始点のみを平滑化処理の対象とする

10

20

30

40

50

ことにより、平滑化処理によって情報が欠損することなく以降の処理を実行することができるため、変倍処理後も局所濃度が保存され、画質劣化を生じることなく、美しい変倍画像を得ることができる。

【第2の実施例】第1の実施例では、予め入力画像全体の画素に対して領域判定処理を行って疑似中間調領域であるか否かを判定し、その結果を領域判定結果蓄積部105に格納したが、この場合、入力画像中の全画素に対応する1ビットのフラグを格納可能なメモリ容量が必要になる。しかし、疑似中間調領域か否かの判定は注目画素の周辺の画素の状態を調べるだけで十分であるため、上記メモリ容量を削減するために、以下のような構成とすることが望ましい。

【0036】即ち、第1平滑化処理を実行する前に領域判定処理を実行するのではなく、第1平滑化処理において各注目ベクトルについて処理を行う際に、入力画像の内、注目ベクトルの始点の周辺領域の画像データのみをアウトライン・ベクトル平滑化／変換部106の不図示の作業用メモリに読込む。そして、注目ベクトルの始点の近傍領域内の各画素について、その画素が疑似中間調領域か否かをその周辺領域内の画素の状態から判定し、近傍領域内で疑似中間調領域と判定された画素の数が所定数 $k$ 個以上であれば、そのベクトルの始点を角点とみなす。尚、前述の周辺領域はベクトル始点の近傍領域内の全画素について疑似中間調領域か否かの判断ができる程度の広さを持つ必要がある。

【0037】図14は、本実施例に係る第1平滑化処理の流れを示すフローチャートである。尚、図14において、第1の実施例と共通するステップについては、同じステップ番号を付している。以下、本実施例に特有のステップS1401乃至S1403について説明する。

【0038】先ず、ステップS1401において、ベクトル用データメモリ104からアウトラインベクトルを読込み、次のステップS1402において、注目ベクトルの始点の周辺領域の画像データのみをアウトラインベクトル平滑化／変換部106の不図示の作業用メモリに読込む。ここで、周辺領域は、先に述べた $N \times N$ の近傍領域内の全画素について、それが疑似中間調領域であるか否かを判定する必要があるため、当然に $N \times N$ の近傍領域を包含する領域である必要がある。

【0039】ステップS1403において、ステップS1401で読込んだベクトルの始点の近傍領域内の各画素についてその画素が疑似中間調領域か否かをその周辺領域内の画素の状態から判定しつつ疑似中間調領域の画素数をカウントし、その結果を変数 $C$ に格納する。そして、以下のステップS1303乃至S1307において、第1の実施例と同様の処理（ステップS1303乃至

S1307）を施すことにより、第1平滑化処理を実行する。

【0040】以上のように、第1平滑化処理において各注目ベクトルについて処理を行う際に、入力画像の内、注目ベクトルの始点の周辺領域の画像データのみをアウトライン・ベクトル平滑化／変換部106の不図示の作業用メモリに読込み、領域判定処理を施すことにより、一連の処理に供するメモリ容量を大幅に削減することができる。

10 【0041】尚、本発明は、複数の機器から構成されるシステムに適用しても、1つの機器から成る装置に適用しても良い。また、本発明はシステム或は装置にプログラムを供給することによって達成される場合にも適用できることはいうまでもない。

【発明の効果】以上説明したように、本発明に係る画像処理方法及び画像処理装置によれば、疑似中間調処理を施された画像の輪郭の平滑化処理における画質劣化が防止される。

【0042】

20 【図面の簡単な説明】

【図1】本実施例に係る画像処理装置の構成を示すブロック図である。

【図2】入力画像からアウトライン・ベクトル抽出する工程を説明する図である。

【図3】階段上のアウトラインベクトル群に対する第1平滑化処理を説明する図である。

【図4】輪郭点を角点とみなす場合の第1平滑化処理を説明する図である。

30 【図5】ノッチを除去する場合の第1平滑化処理を説明する図である。

【図6】入力画像を示す図である。

【図7】入力画像データからアウトライン・ベクトルを抽出した結果を示す図である。

【図8】第1平滑化処理後の輪郭点を示す図である。

【図9】第1及び第2平滑化処理後の輪郭点を示す図である。

【図10】2値画像再生処理を行った結果を示す図である。

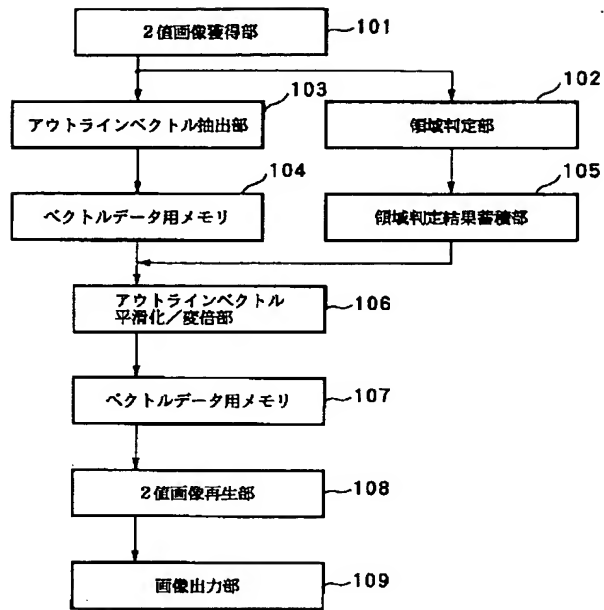
40 【図11】平滑化処理を施さない場合の2値画像を示す図である

【図12】疑似中間調処理された入力画像を示す図である。

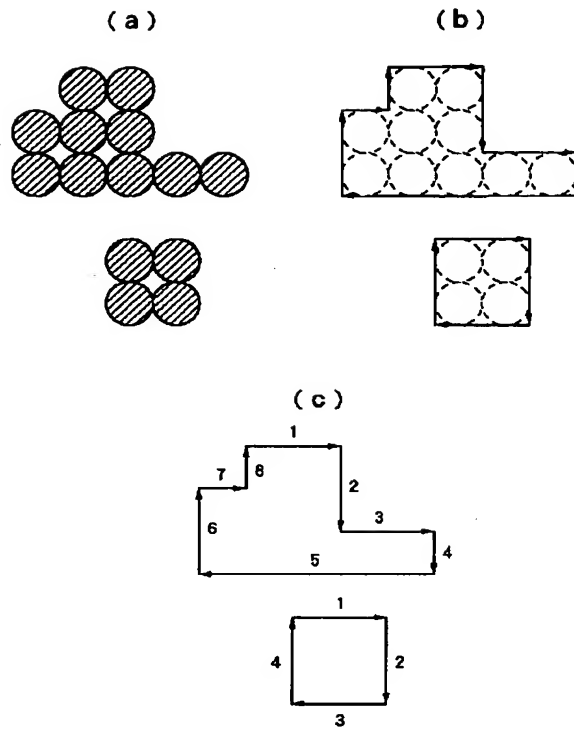
【図13】第1の実施例に係る第1平滑化処理の流れを示すフローチャートである。

【図14】第2の実施例に係る第1平滑化処理の流れを示すフローチャートである。

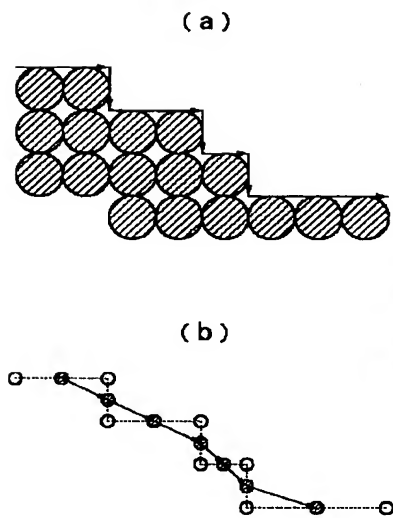
【図 1】



【図 2】

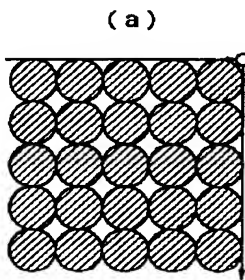


【図 3】

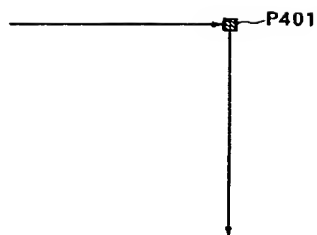


○: 第1平滑化を行う前の輪郭点  
 ⊗: 第1平滑化を行い再設定された後の輪郭点

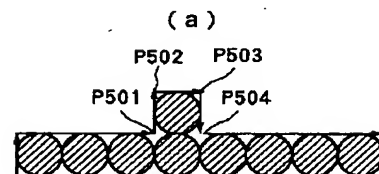
【図 4】



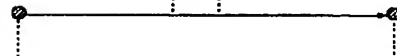
(b)



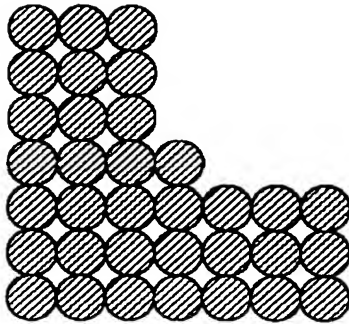
【図 5】



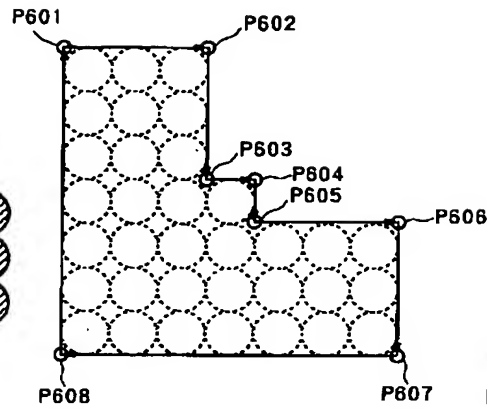
(b)



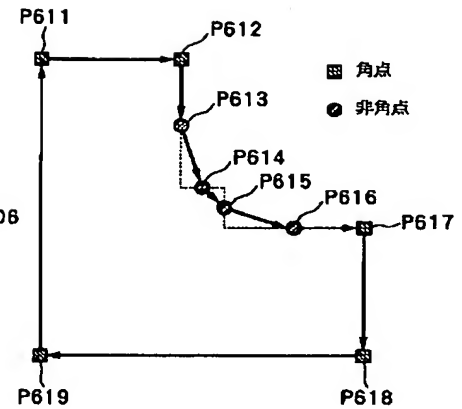
【図6】



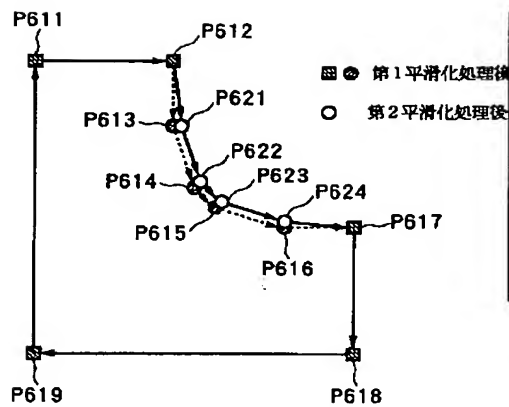
【図7】



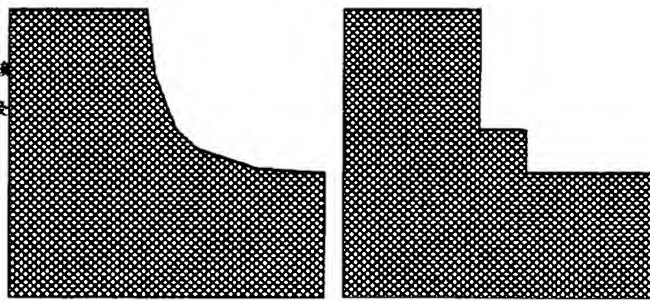
【図8】



【図9】



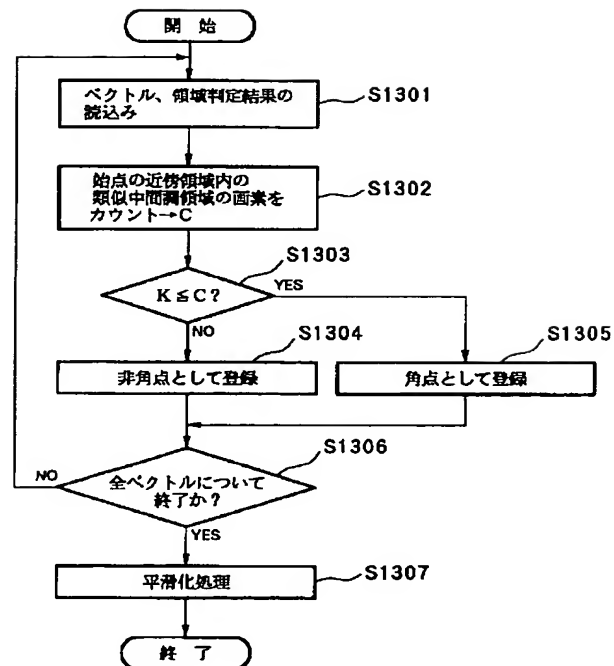
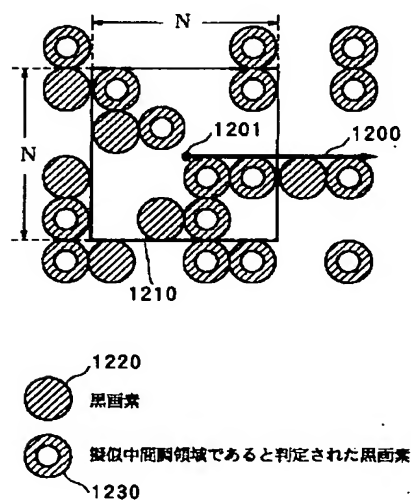
【図10】



【図11】

【図13】

【図12】





【図14】

